

Remarks

[0001] Herein, the "Action" or "Office Action" refers to the Final Office Action dated November 15, 2006.

[0002] Applicant respectfully requests reconsideration and allowance of all pending claims of the application. Claims 1-37 are presently pending. Claims 1, 12, 20, 26, 33, 35 and 36 are amended herein. Claims withdrawn or canceled herein are None. New claims added herein are None.

[0003] Applicant's amendments and remarks after Final are appropriate under 37 C.F.R. §1.116 because they address the Office's remarks in the Final Action, and thus could not have been presented earlier. In addition, the amendments and remarks should be entered to place the case in better form for appeal.

Formal Request for an Interview

[0004] If the Office's reply to this communication is anything other than allowance of all pending claims, then Applicant formally requests an interview with the Examiner of this patent application. I encourage the Examiner to contact me—the undersigned attorney for the Applicant—to schedule a date and time for a telephone interview that is most convenient for both of us. Please email me at chrisf@leehayes.com. Should you contact me by email, please copy my assistant Carly Taylor (carly@leehayes.com) as well. While email works great for me, I welcome you to call either of us as well.

Summary of Muhlestein

[0005] For ease of discussion, Applicant presents a brief summary of Muhlestein. Muhlestein describes that with the increase in the size and the complexity of computer systems and networks, the problem of managing such systems is also increasing. Muhlestein describes that Windows® Management Instrumentation (WMI) is a significant tool that is often used by network developers and administrators to manage computers across an enterprise. Although the WMI enables remote management of windows-based systems and applications by exposing management information through an object-oriented structure, the amount of information exposed by WMI is unwieldy, making it difficult for administrators to efficiently manage their networks. For example, when an administrator wants to accomplish a specific task, such as removing a database, the administrator may have to navigate an extensive schema to locate a particular object and invoke a particular method to accomplish the task.

[0006] Accordingly, Muhlestein describes a command line utility that overlays the WMI technology to provide easier command options for administrators, so that the administrators can more effectively manage their systems and networks (*Muhlestein*, p.2 [0011]). The WMI command line utility provides command options via class aliases that enable mapping of queries, and WMI operations to a common term (*Muhlestein*, p.2 [0011]).

[0007] Muhlestein describes that commands for the WMI command line utility are configured by an underlying object model command schema that defines a mapping between the commands and the WMI schema, such that each command (*i.e.*, class alias within the command schema) defines a target object within the WMI schema against which the command will operate (*Muhlestein*, p.2 [0012]). Muhlestein describes that a class alias is effectively a command that is executed on the command line utility in order to facilitate a specific type of management function (*Muhlestein*, p.5 [0048]-[0052]).

[0008] Muhlestein further describes that the WMI command line utility enables administrators to organize class instances into namespaces (*i.e.*, a logical grouping which reflects the organization of a company's operational environment) in such a way that the command alias required by a specific administrative role can be found together. This enables administrators to focus on the commands which are relevant to their desired tasks, without having to navigate the entire command schema to locate desired command aliases. The network structure is represented as an object-oriented model containing a management station and the target station and the various commands are configured in a class in the model. An input is provided and the input is processed by the commands that communicate with each other through text while processing the input, and once the input has been processed it is formatted and sent to an output command which provides the formatted data for display.

Substantive Claim Rejections

35 USC § 102 Claim Rejections

[0009] Claims 1-37 are rejected under 35 USC §102(e) as being anticipated by U.S. Patent Application Publication No. 2003/0018765 to Muhlestein et al. (hereinafter, "Muhlestein") (*Office Action* p.2).

[0010] Applicant respectfully traverses the rejections, and requests reconsideration and allowance in light of the comments and amendments contained herein. Accordingly, Applicant requests that the rejections be withdrawn and that the case be passed along to issuance.

[0011] **Claim 1** recites a computer-implemented method for processing data, the method comprising:

receiving a parseable object emitted from a prior object-based command within a pipeline comprising a plurality of object-based commands, the prior object-based command being one of the plurality of object-based commands, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with the prior object-based command within the pipeline through the parseable object emitted from the prior object-based command, wherein the parseable object includes at least one method, and wherein an operating environment that supports the pipeline of the plurality of object-based commands is configured to support execution of the object-based commands within the same process;

obtaining a data type for the parseable object;

obtaining format information describing a format for the data type; and

emitting a format object for access by another subsequent object-based command, the format object being based on the format information.

[0012] Applicant initially notes that in the Office Action, the Examiner indicates that claim 1 does not require that the environment comprise the recited features (*Office Action*, p.10). In response, claim 1 has been amended and now recites "receiving a parseable object emitted from a prior object-based command within a pipeline comprising a plurality of object-based commands" and "wherein an operating environment that supports the pipeline of the plurality of object-based commands is configured to support execution of the object-based commands within the same process", thereby making it clear that that pipeline and the environment supporting the pipeline are required features of the claim.

[0013] In order for Muhlestein to anticipate this claim, Applicant submits that Muhlestein must disclose each and every element and feature of the claim and that they must be arranged in the same manner as the claim.

[0014] Applicant respectfully submits that Muhlestein does not show or disclose all of the claimed elements and features of claim 1. For example, Muhlestein does not show or disclose "receiving a parseable object emitted from a prior object-based command within a pipeline comprising a plurality of object-based commands, the prior object-based command being one of the plurality of object-based commands, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with the prior object-based command within the pipeline through the parseable object emitted from the prior object-based command", as recited in claim 1.

[0015] The Office argues that Muhlestein describes "pipelining" of commands and anticipates this claim, and to support this assertion cites to a section of Muhlestein which states that "[a] set of commands for the WMI command line utility is configured by an underlying object model command schema that defines a mapping between the commands and the WMI schema against which the commands will operate" (*Office Action*, pp.3 and 10; *Muhlestein*, p.2 [0012]).

[0016] The Office apparently argues that the “set of commands” described in this section of Muhlestein are pipelined commands which are linked in exactly the same manner recited in claim 1 (*Office Action*, p.10).

[0017] Applicant respectfully disagrees with this interpretation for several reasons described below. First, Muhlestein says nothing about “receiving a parseable object emitted from a prior object-based command within a pipeline comprising a plurality of object-based commands, the prior object-based command being one of the plurality of object-based commands” as recited in claim 1. Instead the “set of commands” referred to in Muhlestein simply describe that the WMI command line utility enables administrators to organize class instances into namespaces (*i.e.*, a logical grouping which reflects the organization of a company’s operational environment) in such a way that the command alias required by a specific administrative role can be found together. Selecting a command from the logical grouping (and entering the command on the command line) simply allows the administrator to focus on the commands which are relevant to his desired tasks, without having to navigate the entire command schema to locate desired command aliases.

[0018] Second, even if the “set of commands” referred to in Muhlestein were a “pipeline”, Muhlestein says nothing about pipelining commands which are linked in exactly the same manner recited in claim 1. More specifically, Muhlestein does not show or disclose that the communication in the pipeline is “such that a subsequent object-based

command within the pipeline which receives the parseable object is configured to communicate with the prior object-based command within the pipeline through the parseable object emitted from the prior object-based command". Instead, Muhlestein describes that the network structure is represented as an object-oriented model containing a management station and the target station and the various commands are configured in a class in the model (Muhlestein, p.2 [0002] p. 4 [0045]-p.7 [0072]). An input is processed by the commands that communicate with each other through text while processing the input, and once the input has been processed it is formatted and sent to an output command which provides the formatted data for display. In other words, Applicant contends that the WMI command line disclosed in Muhlestein reference teaches having each WMI command responsible for receiving text, parsing text, and outputting the information, and does not describe communication between object based commands being accomplished by passing a parseable object.

[0019] Claim 1 also recites "wherein an operating environment that supports the pipeline of the plurality of object-based commands is configured to support execution of the object-based commands within the same process." Traditional command line environments required that each command be an executable and thus, would not execute in the same process. The command line utility disclosed in the Muhlestein reference provides a command line and allows one command to execute a series of API calls. The Muhlestein reference fails to show or disclose an operating

environment that supports the pipeline of the plurality of object-based commands is configured to support execution of the object-based commands within the same process.

[0020] Claim 1 also recites “wherein the parseable object includes at least one method.” Again, as discussed herein, the Muhlestein reference fails to show or disclose emitting a parseable object from a prior command such that a subsequent command within the pipeline receives the parseable object, and the prior command can communicate with subsequent command within the pipeline via the parseable object. In addition, there is no discussion about a parseable object that has at least one method.

[0021] Finally, Claim 1 recites “emitting a format object for access by another subsequent command, the format object being based on the format information.” Again, as discussed herein, the Muhlestein reference fails to teach or suggest emitting any object for access by another subsequent command, and accordingly also fails to teach or suggest emitting a format object.

[0022] Accordingly, claim 1 is allowable over Muhlestein for at least these reasons, and Applicant respectfully requests that the §102 rejection be withdrawn.

[0023] **Claims 2-11** are allowable by virtue of their dependency upon claim 1 (either directly or indirectly). Additionally, some or all of claims 2-11 may be allowable over Muhlestein for independent reasons.

[0024] **Claim 12** recites a computer readable medium including at least one tangible component and having computer-executable instructions for providing data driven output, the instructions comprising:

receiving a parseable object emitted from a prior object-based command within an operating environment that supports a pipeline of a plurality of object-based commands and that is configured to support the execution of the object-based commands within the same process, the prior object-based command being one of the plurality of object-based commands, wherein the receiving occurs as part of the pipeline of the plurality of object-based commands, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with the prior object-based command within the pipeline through the parseable object emitted from the prior object-based command, the parseable object having at least one method;

obtaining a data type for the parseable object;

obtaining format information describing a format for the data type; and

emitting a format object for access by another subsequent object-based command from the plurality of object-based commands, the format object being based on the format information.

[0025] Claim 12 is allowable over Muhlestein based on similar reasoning as discussed above for independent Claim 1, and therefore the Applicant has not repeated the arguments, but rather argues that the same arguments apply as discussed above.

[0026] Accordingly, claim 12 is allowable over Muhlestein for at least the reasons described above and Applicant respectfully requests that the §102 rejection be withdrawn.

[0027] **Claims 13-19** are allowable by virtue of their dependency upon claim 12 (either directly or indirectly). Additionally, some or all of claims 13-19 may be allowable over Muhlestein for independent reasons.

[0028] **Claim 20** recites system that supports data driven output, the system comprising:

a processor;

a memory, the memory being allocated for a plurality of computer-executable instructions which are loaded into the memory for execution by the processor, wherein upon execution of the computer-executable instructions the system being configured to:

receive a parseable object emitted from a prior object-based command within an operating environment that supports a pipeline of a plurality of object-based commands and that is configured to support the execution of the object-based commands within the same process, the prior object-based command being one of the plurality of object-based commands, wherein receiving the parseable object occurs as part of the pipeline, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with the prior object-based command within the pipeline through the parseable object emitted from the prior object-based command, the parseable object having at least one method;

obtain a data type for the parseable object;

obtain format information describing a format for the data type; and

emit a format object for access by a subsequent object-based command from the plurality of object-based commands, the format object being based on the format information.

[0029] Claim 20 is allowable over Muhlestein based on similar reasoning as discussed above for independent Claim 1, and therefore the Applicant has not repeated the arguments, but rather argues that the same arguments apply as discussed above. Accordingly, claim 20 is

allowable over Muhlestein for at least the reasons described above and Applicant respectfully requests that the §102 rejection be withdrawn.

[0030] **Claims 21-25** are allowable by virtue of their dependency upon claim 20 (either directly or indirectly). Additionally, some or all of claims 21-25 may be allowable over Muhlestein for independent reasons.

[0031] **Claim 26** recites a method for providing a data driven command line output, the method comprising:

receiving a command-line instruction containing an output command configured to receive a parseable object, the parseable object having at least one method, wherein the receiving occurs as part of a pipeline of a plurality of object-based commands, such that a subsequent object-based command within the pipeline which receives the parseable object is configured to communicate with a prior object-based command within the pipeline through the parseable object emitted from the prior object-based command; and

executing the output command to manipulate the parseable object and to output a result to an output destination.

[0032] Claim 26 is allowable over Muhlestein based on similar reasoning as discussed above for independent Claim 1, and therefore the Applicant has not repeated the arguments, but rather argues that the same arguments apply as discussed above. Accordingly, claim 26 is allowable over Muhlestein for at least the reasons described above and Applicant respectfully requests that the §102 rejection be withdrawn.

[0033] Claims 27-37 are allowable by virtue of their dependency upon claim 26 (either directly or indirectly). Additionally, some or all of claims 27-37 may be allowable over Muhlestein for independent reasons.

Dependent Claims

[0034] In addition to its own merits, each dependent claim is allowable for the same reasons that its base claim is allowable. Applicant submits that the Office withdraw the rejection of each dependent claim where its base claim is allowable.

Conclusion

[0035] All pending claims are in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the application. If any issues remain that prevent issuance of this application, the Office is urged to contact the undersigned attorney before issuing a subsequent Action.

Respectfully Submitted,

Dated: 4-4-2007

By: 
Christen M. Fairborn
Reg. No. 55,164
(509) 324-9256 x249
christf@leehayes.com
www.leehayes.com